

CLAIM AMENDMENTS

1 1. (Previously Presented) A method of determining a placement of services of a
2 distributed application onto nodes of a distributed resource infrastructure
3 comprising the steps of:
4 forming communication constraints between node pairs which ensure that
5 a sum of transport demands between a particular node pair does not exceed a
6 transport capacity between the particular node pair, each term of the sum
7 comprising a product of a first placement variable, a second placement
8 variable, and the transport demand between the services associated with the
9 first and second placement variables;
10 forming an objective; and
11 employing a local search solution to solve an integer program comprising
12 the communication constraints and the objective, which determines the
13 placement of the services onto the nodes.

1 2. (Previously Presented) A method of determining a placement of services of a
2 distributed application onto nodes of a distributed resource infrastructure
3 comprising the steps of:
4 establishing an application model of the services comprising transport
5 demands between the services;
6 establishing an infrastructure model of the nodes comprising transport
7 capacities between the nodes;
8 forming an integer program that comprises:
9 a set of placement variables for a combination of the services and the
10 nodes, each of the placement variables indicating whether a particular
11 service is located on a particular node;
12 communication constraints between node pairs which ensure that a
13 sum of the transport demands between a particular node pair does not
14 exceed the transport capacity between the particular node pair, each term
15 of the sum comprising a product of a first placement variable, a second
16 placement variable, and the transport demand between the services
17 associated with the first and second placement variables; and
18 an objective; and

19 employing a local search solution to solve the integer program which
20 determines the placement of the services onto the nodes.

1 3. (Canceled)

1 4. (Original) The method of claim 2 wherein the objective comprises
2 minimizing communication traffic between the nodes.

1 5. (Original) The method of claim 2 wherein the application model further
2 comprises processing demands for the services.

1 6. (Original) The method of claim 5 wherein the infrastructure model further
2 comprises processing capacities for the nodes.

1 7. (Original) The method of claim 6 wherein the integer program further
2 comprises processing constraints which ensure that a sum of the processing
3 demands for each of the nodes does not exceed the processing capacity for the
4 node.

1 8. (Original) The method of claim 7 wherein the objective comprises
2 minimizing communication traffic between the nodes and balancing the
3 processing demands on the nodes.

1 9. (Original) The method of claim 6 wherein the processing demands and the
2 processing capacities are normalized according to a processing criterion.

1 10. (Original) The method of claim 9 wherein the processing criterion comprises
2 an algorithm speed.

1 11. (Original) The method of claim 9 wherein the processing criterion comprises
2 a transaction speed.

1 12. (Original) The method of claim 9 wherein the processing capacities of the
2 nodes are found according to a look-up table in which different types of nodes

3 have been normalized according to the processing criterion.

1 13. (Original) The method of claim 2 wherein the application model further
2 comprises storage demands for the services.

1 14. (Original) The method of claim 13 wherein the infrastructure model further
2 comprises storage capacities for the nodes.

1 15. (Original) The method of claim 14 wherein the integer program further
2 comprises storage constraints which ensure that a sum of the storage demands for
3 each of the nodes does not exceed the storage capacity for the node.

1 16. (Original) The method of claim 2 wherein the integer program further
2 comprises placement constraints which ensure that each of the services is placed
3 on one and only one of the nodes.

1 17. (Original) The method of claim 2 wherein the services reside on the nodes
2 according to a previous assignment.

1 18. (Original) The method of claim 17 further comprising the step of assessing
2 reassignment penalties for service placements that differs from the previous
3 assignment.

1 19. (Original) The method of claim 18 wherein the integer program further
2 comprises a second objective that seeks to minimize the reassignment penalties.

1 20. (Previously Presented) A method of determining a placement of services of a
2 distributed application onto nodes of a distributed resource infrastructure
3 comprising the steps of:
4 establishing an application model of the services that comprises processing
5 demands for the services, storage demands for the services, and transport
6 demands between the services;
7 establishing an infrastructure model of the nodes that comprises processing
8 capacities for the nodes, storage capacities for the nodes, and transport

capacities between the nodes;
forming an integer program that comprises:
a set of placement variables for a combination of the services and the nodes, each of the placement variables indicating whether a particular service is located on a particular node;
processing constraints which ensure that a sum of the processing demands for each of the nodes does not exceed the processing capacity for the node;
storage constraints which ensure that a sum of the storage demands for each of the nodes does not exceed the storage capacity for the node;
placement constraints which ensure that each of the services is placed on one and only one node;
communication constraints between node pairs which ensure that a sum of the transport demands between a particular node pair does not exceed the transport capacity between the particular node pair, each term of the sum comprising a product of a first placement variable, a second placement variable, and the transport demand between the services associated with the first and second placement variables; and
an objective of minimizing communication traffic between the nodes and balancing processing loads on the nodes; and
employing a local search solution to solve the integer program which determines the placement of the services onto the nodes.

21. (Previously Presented) A computer readable memory comprising computer code for directing a computer to make a determination of a placement of services of a distributed application onto nodes of a distributed resource infrastructure, the determination of the placement of the services onto the nodes comprising the steps of:

forming communication constraints between node pairs which ensure that a sum of transport demands between a particular node pair does not exceed a transport capacity between the particular node pair, each term of the sum comprising a product of a first placement variable, a second placement variable, and the transport demand between the services associated with the first and second placement variables;

12 forming an objective; and
13 employing a local search solution to solve an integer program comprising
14 the communication constraints and the objective, which determines the
15 placement of the services onto the nodes.

1 22. (Previously Presented) A computer readable memory comprising computer
2 code for directing a computer to make a determination of a placement of services
3 of a distributed application onto nodes of a distributed resource infrastructure, the
4 determination of the placement of the services onto the nodes comprising the steps
5 of:

6 establishing an application model of the services comprising transport
7 demands between the services;

8 establishing an infrastructure model of the nodes comprising transport
9 capacities between the nodes;

10 forming an integer program that comprises:

11 a set of placement variables for a combination of the services and the
12 nodes, each of the placement variables indicating whether a particular
13 service is located on a particular node;

14 communication constraints between node pairs which ensure that a
15 sum of the transport demands between a particular node pair does not
16 exceed the transport capacity between the particular node pair, each term
17 of the sum comprising a product of a first placement variable, a second
18 placement variable, and the transport demand between the services
19 associated with the first and second placement variables; and

20 an objective; and

21 employing a local search solution to solve the integer program which
22 determines the placement of the services onto the nodes.

1 23. (Canceled)

1 24. (Original) The computer readable memory of claim 22 wherein the objective
2 comprises minimizing communication traffic between the nodes.

1 25. (Original) The computer readable memory of claim 22 wherein the

2 application model further comprises processing demands for the services.

1 26. (Original) The computer readable memory of claim 25 wherein the
2 infrastructure model further comprises processing capacities for the nodes.

1 27. (Original) The computer readable memory of claim 26 wherein the integer
2 program further comprises processing constraints ensure that a sum of the
3 processing demands for each of the nodes does not exceed the processing capacity
4 for the node.

1 28. (Original) The computer readable memory of claim 27 wherein the objective
2 comprises balancing the processing demands on the nodes.

1 29. (Original) The computer readable memory of claim 26 wherein the processing
2 demands and the processing capacities are normalized according to a processing
3 criterion.

1 30. (Original) The computer readable memory of claim 29 wherein the processing
2 criterion comprises an algorithm speed.

1 31. (Currently Amended) The computer readable memory of claim 29 wherein
2 the processing criterion comprises a transaction speed.

1 32. (Currently Amended) The computer readable memory of claim 29 wherein
2 the processing capacities of the nodes are found according to a look-up table in
3 which different types of nodes have been normalized according to the processing
4 criterion.

1 33. (Original) The computer readable memory of claim 22 wherein the
2 application model further comprises storage demands for the services.

1 34. (Original) The computer readable memory of claim 33 wherein the
2 infrastructure model further comprises storage capacities for the nodes.

1 35. (Original) The computer readable memory of claim 34 wherein the integer
2 program further comprises storage constraints which ensure that a sum of the
3 storage demands for each of the nodes does not exceed the storage capacity for the
4 node.

1 36. (Original) The computer readable memory of claim 22 wherein the integer
2 program further comprises placement constraints which ensure that each of the
3 services is placed on one and only one of the nodes.

1 37. (Original) The computer readable memory of claim 22 wherein the services
2 reside on the nodes according to a previous assignment.

1 38. (Original) The computer readable memory of claim 37 further comprising the
2 step of assessing reassignment penalties for service placements that differs from
3 the previous assignment.

1 39. (Original) The computer readable memory of claim 38 wherein the integer
2 program further comprises a second objective that seeks to minimize the
3 reassignment penalties.

1 40. (Previously Presented) A computer readable memory comprising computer
2 code for directing a computer to make a determination of a placement of services
3 of a distributed application onto nodes of a distributed resource infrastructure, the
4 determination of the placement of the services onto the nodes comprising the steps
5 of:

6 establishing an application model of the services that comprises processing
7 demands for the services, storage demands for the services, and transport
8 demands between the services;

9 establishing an infrastructure model of the nodes that comprises processing
10 capacities for the nodes, storage capacities for the nodes, and transport
11 capacities between the nodes;

12 forming an integer program that comprises:

13 a set of placement variables for a combination of the services and the
14 nodes, each of the placement variables indicating whether a particular

15 service is located on a particular node;
16 processing constraints which ensure that a sum of the processing
17 demands for each of the nodes does not exceed the processing capacity for
18 the node;
19 storage constraints which ensure that a sum of the storage demands for
20 each of the nodes does not exceed the storage capacity for the node;
21 placement constraints which ensure that each of the services is placed
22 on one and only one node;
23 communication constraints between node pairs which ensure that a
24 sum of the transport demands between a particular node pair does not
25 exceed the transport capacity between the particular node pair, each term
26 of the sum comprising a product of a first placement variable, a second
27 placement variable, and the transport demand between the services
28 associated with the first and second placement variables; and
29 an objective of minimizing communication traffic between the nodes
30 and balancing processing loads on the nodes; and
31 employing a local search solution to solve the integer program which
32 determines the placement of the services onto the nodes.